REMARKS

Claims 1 and 3-4 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Croslin, further in view of Beurket et al., further in view of Swildens et al., U.S. Patent No. 6,484,143. Claim 2 has been rejected under 35 U.S.C. § 103(a) as being unpatentable over Croslin, further in view of Beurket et al. and Swildens et al., further in view of Guenthner et al., U.S. Patent No. 6,360,262. Claims 5-7 were rejected "for similar reasons."

Respectfully, these rejections are traversed.

In considering the most recent rejection, Croslin is said to teach "a method of mapping the Internet to generate an optimized set of proxy points <u>in a local name server address space</u>" in part by "executing a route over the Internet from each data center <u>to a given local name server</u>." This is not the case. In particular, each pending claim concerns testing an actual network with respect to an Internet "name server." Claim 1 emphasizes this, e.g., by reference to "a local name server address space" and executing trace routes "to a given local name server." Likewise, claim 5 refers to "end user local name server requests" in the preamble and requires that the routes are physically traced from each mirror site "to the local name server." Claim 7 states that the "client request" is a "client name server request," and this claim also states that the routes being run intersect at a "given name server." Croslin, which concerns a voice/data telecommunications network, says nothing about Internet name servers, name server address space maps, name server locations, trace routing to name servers, or the like. These "name server" limitations in each claim are meaningful, and they are neither disclosed nor suggested in Croslin.

Further, as previously pointed out, the present invention involves <u>physical probing</u> of <u>physical devices on a network</u> to generate relevant data; the portions of Croslin relied upon by the Examiner involve software-driven analysis <u>of a database table</u> that allegedly illustrates a set of physical telecommunications links. Thus, in the present invention, a relevant method step calls for physical acts to take place over a physical network; at most, Croslin simply teaches analyzing a "representation" of a physical network, not performing the relevant tests on the physical network itself. Moreover, the representation in Croslin has nothing to do with a "local name server address space" let alone "executing a route over the Internet from [each] data center to a given local name server."

As previously noted, the claims describe this physical probing to physical name servers explicitly:

"for a given pair of data centers each accessible over the Internet, physically executing a trace route over the Internet from each data center to a given local name server" (claim 1)

"for each local name server, physically directing a trace route over the public Internet from each content provider mirror site to the local name server" (claim 5);

Independent claim 7 is even more specific about the physical probing:

"dynamically determining a set of proxy points, wherein each proxy point of the set of proxy points is determined by physically directing a trace route over the public Internet from each of the set of mirror sites toward a given name server and determining a given point in the public Internet where the trace routes from each of the set of mirror sites intersect."

The above-cited claim limitations illustrate the physical nature of the method step at issue and further emphasize that the steps are taken with respect to given name servers; these aspects of the present invention are completely absent from Croslin.

In particular, Croslin simply describes a system 200 that runs a restoration process 208, which is a software routine. The process 208 reads and writes data from a restoration database 212. This data includes a series of tables that represent the network topology and are used to develop the restoration routes. As described in Figure 5, there are a series of steps performed to generate a restoral route for a given impacted trunk of the telecommunications network. Initially, an "intersection table" is built that identifies instances where a sub-route that originates from a given "lefthand" node intersects (i.e. shares at least one common node) with a sub-route that originates from a "righthand" node. A lefthand node is one that lies to the left of the network outage and a righthand node is one that lies to the right of the network outage. According to the patent, the lefthand node is "folded out" to identify from the sub-route table all nodes that are end nodes from a sub-route that begins from the node being folded out. Once the intersection table (see Figure 10) for the impacted trunk is generated, the table can be read to determine the restoral route. In particular, the restoration process 208 checks whether an intersection is found in the intersection table. If so, an optimal intersection route is selected, and this is typically the route with the lowest cost. The selection of the route constitutes a selection or combination of the two end node pairs that intersect, as indicated by the data in the table.

The written description of the present invention relates to a technique for generating a network map for a user-base of the Internet. Instead of probing each local name server that is connectable to a mirrored data center, however, the network map identifies connectivity with respect to a much smaller set of points, referred to in the application as "core" or "common" points. Each set of mirrored data centers preferably has an associated map that identifies a set of core points. In one embodiment, the core points are identified using an incremental trace route that is physically executed (i.e., run) from each of the set of mirrored data centers to a local name server. A "trace route" (or "traceroute") is a computer network test used to determine the route taken by data packets across an actual IP network. This is a real test carried out on a real network with respect to a real name server, not simply a software instruction executed against a database representation, as in Croslin. An intersection of the trace routes at a common routing point is then identified. The core point discovery process is illustrated in Figure 3. Preferably, the trace routes are between data centers and local name servers. The core points are the intersection point of a trace route or near or substantially near the intersection point.

Unlike the present invention, Croslin does not locate intersection points utilizing trace routes performed on the actual Internet. Croslin builds sub-route tables based upon nodes identified by topology data. Points are identified from these database tables, and not from a trace route that is actually performed in a network being mapped.

Neither Buerket et al. nor Guenthner et al., two of the secondary references, make up for the deficiencies in Croslin. In particular, neither reference discloses (nor does the Examiner say otherwise) the physical probing of a physical network to facilitate generation of a proxy point map according to the present invention.

Swildens et al., the newly-cited reference, is of general interest to the present invention (and it is commonly-owned by the assignee of this application). While this patent teaches global load balancing across mirrored origin sites and the use of trace routing, the subject claims are very specific to a map making method that is neither disclosed nor remotely suggested by Swildens et al.

Under Section 103, it is the subject matter "as a whole" that must be considered for patentability. Even if the references could be combined in the manner proposed by the Examiner, at least the following limitations are absent:

"for a given pair of data centers each accessible over the Internet, physically executing a trace route over the Internet from each data center to a given local name server" (claim 1)

"for each local name server, physically directing a trace route over the public Internet from each content provider mirror site to the local name server" (claim 5);

"dynamically determining a set of proxy points, wherein each proxy point of the set of proxy points is determined by physically directing a trace route over the public Internet from each of the set of mirror sites toward a given name server and determining a given point in the public Internet where the trace routes from each of the set of mirror sites intersect (claim 7)

To further emphasize that the method of the present invention is associated with building a name server map, each independent claim has been amended to further clarify that the name server is one of a plurality of name servers that clients use to access resources on the Internet. This language is not required in view of the prior art but is added to provide additional clarity as to the particular environment in which the inventive method is designed to operate.

All claims should be allowed, and a notice to that effect is respectfully requested.

Respectfully submitted,

David H. Judson
Registration No. 30,467
(972) 385-2018

Date: December 26, 2006